基于BIM和本体的建筑运维管理研究

陈贵涛[1,2]

(1. 深圳市建筑科学研究院股份有限公司,深圳 518049; 2. 哈尔滨工业大学深圳研究生院,深圳 518055)

【摘 要】为促进建筑运维管理在语义网上的实现,建立了基于BIM和本体技术的智能应用框架;结合DogOnt本体建立了建筑运维管理本体;提出了BIM数据提取方法及个体、规则的创建方法;采用Revit API、OWL API和SWRL API技术实现了框架程序。利用一个实际项目验证了框架、方法和程序的有效性。对推进BIM、本体和语义网技术的发展以及建筑运维的网络化、智能化管理具有重要意义。

【关键词】建筑运维管理;本体;BIM;OWL;SWRL
【中图分类号】TU17 【文献标识码】A 【文章编号】1674-7461(2017)04-0067-07
【DOI】10.16670/j.cnki.cn11-5823/tu.2017.04.13

建筑生命周期内,运维成本可能数倍于建造成本[1]。有效建筑运维管理以准确详尽的运维信息为基础,是一种高效智能的信息化管理。然而,目前建筑行业普遍存在的信息孤岛、低效的手动操作方式以及运维数据的非数字化等问题已成为实现有效建筑运维管理的主要阻碍。

建筑信息模型(BIM)是"一个新兴的设计、施工和运维管理方法,该方法采用数字化形式表达建设过程,并基于这种数据模型实现建筑信息交互。"[2]。BIM技术在建筑运维领域内的应用,可实现协调、一致和可量化的运维管理。然而,BIM数据与运维管理系统间的交互是一个低效的易于出错的过程。同时,BIM数据存储技术标准—IFC标准[3]缺少语义,限制了BIM数据的智能应用[4]。

本体作为共享概念模型的明确的形式化规范说明,表达了领域内若干概念之间的内在关系[5]。本体可以提供共同的术语和形式化的语义,能够进行一致性检查,实现异质数据的集成和不同应用间的互操作,并可实现领域本体的复用和推理[6]。更重要的是,本体技术可提供一种机器可理解的数据语义,是语义网[7]对事物的描述基础。建立建筑运维的本体化描述,对实现运维的网络化、智能化管理具有重要作用。

本文实现了基于BIM与本体技术的建筑运维管理的语义描述与智能推理,为建筑运维管理的语义网实现提供了重要基础,对BIM技术、本体语义网技术的发展及建筑运维的智能化、网络化管理起重要促进作用。

1 建筑运维管理架构

基于BIM和本体的智能应用框架如图1所示。OWL(Web Ontology Language,网络本体语言)[8]知识库通过构建建筑本体与其他领域本体的关联,按建筑本体TBox提供的信息结构,从BIM模型中提取建筑信息,建立建筑本体的ABox。SWRL(Sematic Web Rule Language,语义网规则语言)[9]规则引擎桥用于OWL知识库和规则引擎的交互,实现规则推理和OWL知识检索。框架中本体管理系统用于创建、编辑本体。推理机可执行异质监测、TBox计算等任务,以确保本体的一致性。

图1框架的实现需要依靠Revit API、OWL API[10]和SWRL API[9]三个应用程序编程接口。OWL API的实体和类的表述以及公理的接口层级和名称与OWL结构标准直接对应。这种规定允许开发者在一个合适的抽象水平上工作,从复杂的本体语法描述中脱离出来。同时,OWL API提供了推

【基金项目】 国家重点研发计划课题(项目编号:2016YFC0702016)
【作者简介】 陈贵涛(1983-),男,博士,主要从事建筑信息化、结构优化、BIM等方面的研究。

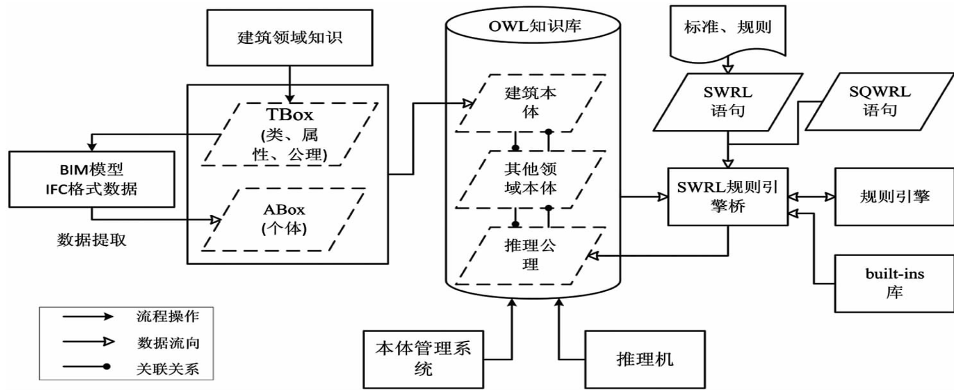


图 1　基于 BIM 和本体的智能应用框架

理机接口，允许推理功能独立于本体知识库运行，减轻了本体的负担与复杂度。

SWRL API 采用规则引擎实现了 OWL 知识库、SWRL 规则和第三方规则引擎之间的交互。逻辑以规则的形式独立于 OWL 知识库被推理和处置。另外，SWRL API 支持 built-ins 实现用户定义的用于规则中的判断，同时 SWRL API 提供了 built-ins 桥和一系列 built-ins 库，这对于拓展 SWRL 非常有利。另外 SWRL API 采用 SWQRL[11] 实现 OWL 知识库的检索。SQWRL 采用标准 SWRL built－in 机制，作为一个 built－in 库执行，规则推理与本体检索采用统一的方式执行。

这样，用户从底层的语法、语义、逻辑的数据操作中脱离出来，专注于 ABox、SWRL 规则和 SQWRL 查询等应用层面上的工作，提升了应用框架的适用性，对于拓展本体技术的推广具有重要意义。同时，BIM 技术也被融入应用框架，随着语义网技术的发展，建筑信息将不再是存储在专业图纸中的专业信息，而是可以被 Web 访问，被机器引擎自动处理的有效信息。建筑信息将与其他信息一起构成语义丰富、相互关联的知识库，建筑信息的应用潜力将是不可估量的。

2　程序结构

实现图 1 应用框架的程序结构如图 2 所示。程序采用 Revit API 技术，基于 Revit 软件[12]（BIM 系统）采用 C#程序语言实现。系统主要由五部分组成，BIM 系统提供建筑运维信息，图形化展示运维结果；数据库主要用于存储建筑运维数据和本体推理结果；建筑运维管理本体是包含了个体和 SWRL 规则库的 OWL 知识库，是建筑运维智能管理和网络管理的核心。

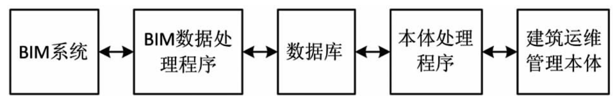


图 2　程序结构

上述三部分通过 BIM 数据处理程序和本体处理程序实现数据交互。BIM 数据处理程序一方面可按照建筑运维管理本体对建筑数据的结构约束，从 BIM 模型中自动提取数据并存储至数据库。另一方面，该程序可自动读取数据库中存储的本体推理结果，实现运维管理所需的图形展示功能。另外，建筑运维管理功能的流程控制也通过该程序实现。本体处理程序采用 Java 技术建立，可实现对本体的创建、更新以及对规则和查询的建立和执行。另外该程序可将推理结果存储至数据库，以便 BIM 处理程序调用。

3　建筑运维本体

本质上，建筑运维本体提供一个信息结构，能准确完备地描述建筑运维的信息需求。如表 1 所示，建筑运维的信息包括几何信息和非几何信息两类[13]。

本研究以 DogOnt[14] 本体为基础，拓展了 TBox，建立了建筑运维本体。一些基本的概念和关系如图 3 所示。其中，Building Thing 用于模拟建筑构件、设施和设备；Building Environment 用于模拟建筑的空间信息；Sate 用于模拟可控单元可以呈现的稳定状态；Functionality 用于模拟可控单元的功能；Network Component 用于模拟网络特征；Staff 用于模

拟运维人员。这些基本概念又被详细分为若干子类。例如 State 按状态特征被划分为连续值(Continue State)和离散值(Discrete State)。

基本概念通过对象属性建立关联,例如 Control 类中的建筑设备个体通过 hasFunction 与 Function 中的某个功能个体关联。同时概念与其特征描述通过数据属性建立关联,例如 Room 类中的办公室个体通过 hasRoomArea 与 xsd:double 类型的数据关联。由于本体 TBox 包含的概念和关系较多,限于篇幅限制,有关 DogOnt 本体的详细介绍参见参考文献[14]。

表 1　建筑运维的信息需求

类别	子类别	信息需求
几何信息	建筑信息	建筑构件、结构构件的几何信息;
	设备设施信息	机械、电力、管道、通信和消防系统的几何信息;
	空间信息	建筑、楼层、房间和区域信息;构件、设备和设施的空间信息;
	空隙信息	构件、设备和设施的空隙信息;
非几何信息	属性	构件的编号、名称、分组、材料信息; 设备和设施基本属性:编号、采购、保修、备件、使用情况等信息; 设备和设施状态属性:运行状态、运行历史、维护历史、限值、功率、能耗等信息;
	文件	设备设施的合格证、保修文件、操作和使用说明等; 管理制度;工作程序等;
	人员	姓名、分组、职责、薪酬、工作状态等信息

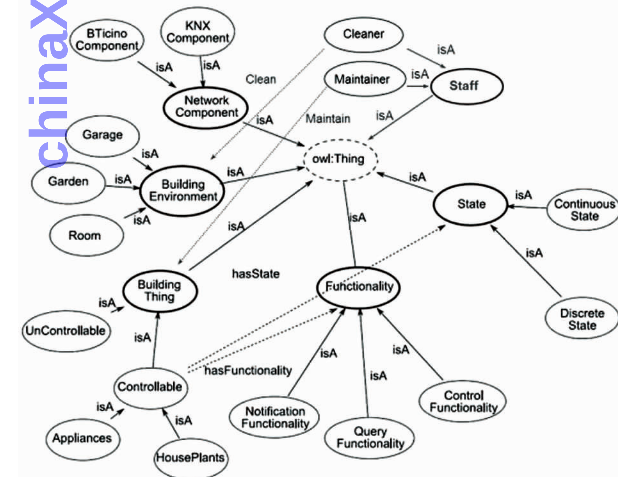


图 3　建筑运维本体片段

4　BIM 信息提取方法

BIM 模型集成了建筑生命周期内的信息,作为建筑信息数据库为不同阶段不同利益相关者提取需求信息提供了数据基础。然而,从 BIM 模型中提取信息是一个冗繁的过程。有学者基于 IFC 文件和 ifcXML 文件建立提取程序提取需求信息[12, 15]。这种方法存在两个缺点:一是 IFC 和 ifcXML 文件中实体、单元、属性和数值间通过复杂的 ID 引用实现,为找到某个单元的相关信息需要循环遍历大量文本,信息提取效率低;二是尽管基于 IFC 和 ifcXML 文档可以获得大部分建筑单元的信息和简单的关联关系,一些复杂的关联关系和建筑空间信息是无法获取的。

本文采用 RevitAPI 技术根据 TBox 中关于信息的要求与约束,从 BIM 模型中直接提取相关信息,建立个体。

4.1　个体的提取

个体信息的提取首先应建立一个 RevitAPI 参数到本体 TBox 的映射,如表 2 所示。

表 2　RevitAPI 参数到本体 TBox 的映射片段

Revit API 参数	类型	运维本体类
Room	基类	Room
OST_Floors	族类型	Storey
OST_Walls	族类型	Wall
OST_Windows	族类型	Window
OST_Doors	族类型	Door
OST_ElectricalFixtures	族类型	ElectricalSystem
OST_LightingDevices	族类型	Lighting
OST_PlumbingFixtures	族类型	SanitaryFixtures
OST_DataDevices	族类型	DataDevice
OST_MechanicalEquipment	族类型	HVACSystem
OST_PipeAccessory	族类型	PipeAccessor
OST_DuctTerminal	族类型	DuctTerminal
OST_FireAlarmDevices	族类型	FireAlarmDevices
OST_LightingFixtures	族类型	Button

创建逻辑的不同导致运维本体与 IFC 范例无法完全一致。RevitAPI 中对建筑的分类细度较大。例如 OST_MechanicalEquipment 是水泵、压缩机等机械设备的内置类别参数,然而在运维本体内水泵和压缩机是两个类。因此,在映射中将 RevitAPI 参数映射至运维本体较高层级的类上,再根据具体单元的其他属性信息确定其归属。

以 Room 类为例,可利用 RevitAPI 的 Room 参数获取建筑内的所有房间,再根据房间的名称确定其类的归属。如果房间名称中包含“办公”则属于 Office 类,如果包含“卫生间”或“厕所”则属于 RestRoom 类。

4.2　个体属性的提取

数据属性的提取可通过 RevitAPI 内置参数或共享参数直接从 BIM 模型中提取,前提是建立数据

属性与内置参数或共享参数的映射。以房屋的数据属性为例，数据属性与 RevitAPI 参数间的映射关系如表 3 所示。

个体间的对象属性涉及他们的空间、逻辑等无法被 IFC 直接表示的复杂关系。需要分别针对对象属性表示的含义，从 BIM 模型中提取相关信息逐个建立。同时，在本体内数据属性和对象属性还具有属性约束，在实现过程中应满足属性约束。

表 3　RevitAPI 参数到本体数据属性的映射片段

数据属性	RevitAPI 参数	参数属性
hasRoomID	ROOM_NUMBER	BuiltInParameter
hasRoomName	ROOM_NAME	BuiltInParameter
hasRoomArea	ROOM_AREA	BuiltInParameter
hasRoomOccupancy	ROOM_OCCUPANCY	BuiltInParameter
hasTenancyDepartment	ROOM_Department	SharedParameter
hasWarning	ROOM_Warning	SharedParameter
hasTenancyEndTime	ROOM_TenancyEndTime	SharedParameter
hasTenancyStartTime	ROOM_TenancyStartTime	SharedParameter
hasTenancyTerm	ROOM_TenancyTerm	SharedParameter

```
functionbuildingInfo = buildingInfoExtract(para)
if (para 是内置参数或共享参数)
利用 revit API 函数直接获取 buildingInfo
else if (para 是与空间相关的参数)
获得 BIM 模型中的所有空间 rooms
获得 para 对应对象 object
foreach(room in rooms)
    获得空间边界 boundary
    获得 object 的位置 location
    if(location in boundary)
判定 object 在空间 room 内
    endif
endforeach
endif
```

图 4　信息提取伪码示例

从 BIM 模型中提取信息的程序伪码如图 4 所示。需要说明的是 RevitAPI 中给出了设施、设备族实例的 Room 属性，用于表示其所属的房间。但这一属性经常会因为建模、操作等原因结果为 null。图 4 所示程序通过空间信息判断设施设备的空间属性，更具鲁棒性。

4.3　本体个体、规则和检索的建立

本研究中，本体个体、规则和检索的建立采用本体处理程序实现。本体处理程序中采用 OWLAPI 提供的在一定抽象水平上的接口实现个体声明、类声明和属性声明。个体是类的实例，应遵循类的层级和属性约束，因此，这些声明应该是遵循 TBox 中关于类的描述，是一种定制的方法。另外，本体技术遵循开放世界假设 – 没有显式说明的信息就是未知，在个体信息更新过程中，应满足 TBox 关于属性的存在性约束和基数约束。对于具有唯一对象的属性的更新，应删除旧的声明再建立新的声明。

规则和检索的建立采用 SWRL 语言实现。SWRL 语法包含两个主要部分，前提(antecedent)和结果(consequent)，这两部分用符号“→”相连。“前提”和“结果”都是元素(atom)的集合体，各元素间采用“^”相连。SWRL 提供了 7 中类型的元素：类元素，个体属性元素，数据值属性元素，区别个体元素，相同个体元素，built – in 元素和数据取值范围元素。每个元素中的变量以“?”开头附以字母表示。一个类元素由本体内一个类的名称和该类的一个个体或变量组合而成。一个个体属性元素由一个对象属性和两个变量表示的个体组成。相似的，一个数值属性元素由一个数值属性和两个变量组成，第一个变量表示 OWL 个体，第二个变量表示数据属性或值。区分个体元素和相同个体元素用于区分两个变量是否为同一个个体。built – in 元素是 SWRL 提供的最先进的特征，它能够提供更复杂的谓语描述，包括数学计算。

5　应用案例

选择一个实际项目 – 未来立方的空间(图 5)为例验证研究本文提出的方法。该项目建筑面积 658.18m^2，建筑高度 10.2m。采用 9m × 9m × 9m 的空间布局，空间内部功能分区采用模块化技术实现，可根据需求自由转换。



图 5　未来立方项目

采用 SWRL 规则语言建立一个推理规则：如果房间 c 的租期还有 1 个月到期，则对房间 c 进行预警。推理规则的 SWRL 描述如下：

Rule1：abox：dpaa(?c，hasTenancyEndTime，?t)^

temporal:duration(?d,“now”,?t,“Months”)^
swrlb:lessThan(?d, 2)→hasWarning(?c,“Y”)

BIM 数据处理程序提供规则和检索语句界面，当规则被要求执行时，BIM 数据处理程序会根据 TBox 提取空间信息(如图 6 所示)，并将空间信息存储至数据库。之后程序会调用封装了本体处理程序的 JAR，建立 ABox(如图 7 所示)。随后规则将被 SWRL 规则桥导入至规则引擎，dpaa、duration 和 lessThan 三个库内的 build-ins 将通过 SWRL built－in 引擎桥导入至规则引擎。规则引擎进行语义推理，推理得出的新的断言将通过 SWRL 规则引擎桥插入 OWL 本体。

规则引擎将找到满足条件的房间，并遵循开放世界假设给这些房间增加一个断言(c hasWarning “Y”)。因此在增加断言之前，这些房间原有的 hasWarning 属性应被删除。

查找这些房间的查询语句可描述为：

Q1:abox:dpaa(?c,hasTenancyEndTime,?t)^
temporal:duration(?d,“now”,?t,“Months”)
^swrlb:lessThan(?d, 2)→ swrlb:select(?c)

这些房间的数据将被存储至数据库，BIM 处理程序将从数据库中读取结果信息，并将满足条件的房间的“续租预警”属性更新为 ture，并打开各房间在 BIM 软件中对应的平面视图与三维视图，实现推理结果的直观展示。

实际项目中，没有房间到期，为验证方法有效性做如下调整：F1N005 的租期调整为 1 年，开始时间和结束时间分别为 2016－03－01 和 2017－03－01。执行推理结果如图 7 和图 8 所示。可以看出，在 OWL 本体中，F1N005 的 hasTenancyStartTime、hasTenancyEndTime、hasWaring 属性对应数据分别为 2016－03－01、2017－03－01 和 Y。在Revit 软件中，对应房间的视图已经打开，属性也已修改。

为进一步验证个体信息提取方法，按图 4 的流程，以 Revit 单元内置 ID 为主键，提取了房屋内消防、机械、电力、空调等设备信息。截取信息片段如图所示，以房间 2376343 为例，设备分布如图 9 和图 10 所示。可以看出，本文方法可以有效利用空间关系，处理对象属性公理的构建。

提取BIM模型中的房间信息　提取　导出表格

编号	名称	标高	上限	标示高度	面积	占用	租期	租用单位	开始时间	到期时间	续租预警
B1N001	真空泵房	B1	F1J1	2700	12.76	85%					
F1J1N001	卫生间	F1J1	F2	3000	19.77	100%					
F1N001	卫生间	F1	F1J1	1500	10.81	100%					
F1N002	楼梯间	F1	F2	4500	12.61	100%					
F1N003	电梯间	F1	F3	9000	9.68	100%					
F1N004	办公区	F1	F2	4500	77.02	10%	5年	低碳城运营中心	2016-10-01	2021-10-01	
F1N005	展示区	F1	F2J1	6000	89.00	15%	5年	低碳城运营中心	2016-10-01	2021-10-01	
F1N006	外廊	F1	F2	4500	50.95	0%					
F1N007	弱电间	F1	F2	4500	1.56	100%					
F1N008	强电间	F1	F2	4500	1.20	100%					
F1N009	弱电控制室	F1	F2	4500	8.22	100%					
F2J1N002	楼梯间	F2J1	F3	3000	11.23	100%					
F2J1N003	门廊	F2J1	F3	3000	16.94	0%					
F2J1N005	小型办公展示区	F2J1	F3	3000	56.00	10%	3年	深圳市龙岗区...	2016-10-01	2019-10-01	
F2N001	卫生间	F2	F3	4500	7.84	100%					
F2N002	卫生间	F2	F3	4500	9.72	100%					
F2N003	楼梯间	F2	F3	4500	16.76	100%					
F2N004	强电井	F2	RF1	8150	0.54	100%					
F2N005	弱电井	F2	RF1	8150	0.72	100%					
F2N006	开放活动区	F2	F3	4500	114.37	60%	3年	深圳市龙岗区...	2016-10-01	2019-10-01	
F2N007	外廊	F2	F3	4500	50.95	0%					
F2N008	楼梯间	F2	F3	4500	9.83	100%					
F3N001	FLEXLAB实验室	F3	RF1	3650	27.47	0%					
F3N002	FLEXLAB实验室	F3	RF1	3650	28.81	5%					
F3N003	机房	F3	RF1	3650	7.40	60%					
F3N004	设备间	F3	RF1	3650	7.52	5%					
F3N005	设备间	F3	RF1	3650	8.29	5%					
F3N006	设备放置区	F3	RF1	3650	142.76	8%					
RFN001	屋顶	RF	RF3	2200	24.91	0%					
RFN002	太阳能屋顶	RF	RF3	2200	46.87	100%					

图 6　BIM 信息提取方法

```
<!-- http://www.semanticweb.org/chenguitao/ontologies/FMOnt#F1N005 -->

<owl:NamedIndividual rdf:about="http://www.semanticweb.org/chenguitao/ontologies/FMOnt#F1N005">
    <rdf:type rdf:resource="http://www.semanticweb.org/chenguitao/ontologies/FMOnt#ExhibitionRoom"/>
    <rdf:type rdf:resource="http://www.semanticweb.org/chenguitao/ontologies/FMOnt#Room"/>
    <localedIn rdf:resource="http://www.semanticweb.org/chenguitao/ontologies/FMOnt#F1"/>
    <roomUpLimitIs rdf:resource="http://www.semanticweb.org/chenguitao/ontologies/FMOnt#F2J1"/>
    <hasHeight rdf:datatype="http://www.w3.org/2001/XMLSchema#double">6000.0</hasHeight>
    <hasRoomArea rdf:datatype="http://www.w3.org/2001/XMLSchema#double">89.0</hasRoomArea>
    <hasRoomID rdf:datatype="http://www.w3.org/2001/XMLSchema#string">F1N005</hasRoomID>
    <hasRoomName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">展示区</hasRoomName>
    <hasRoomOccupancy rdf:datatype="http://www.w3.org/2001/XMLSchema#double">15.0</hasRoomOccupancy>
    <hasTenancyDepartment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">低碳城运营中心</hasTenancyDepartment>
    <hasTenancyEndTime rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2017-03-01</hasTenancyEndTime>
    <hasTenancyStartTime rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2016-03-01</hasTenancyStartTime>
    <hasTenancyTerm rdf:datatype="http://www.w3.org/2001/XMLSchema#string">1年</hasTenancyTerm>
    <hasWarning rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Y</hasWarning>
</owl:NamedIndividual>
```

图 7　推理后 F1N005 房间的本体描述片段



图 8　推理结果的 BIM 展示

2236994	2236996	2237000	2237002	2238790	2239991	2239993	2376343
2295112	2295119	2295322	2322876	2295320	2295082	2295092	2295134
2295113	2295121	2295265	2323054	2295149	2295084	2295094	2295136
2295124	2295122	2295366		2295266	2295088	2295096	2295137
2295125	2295123	2295367		2295363	2295090	2295098	2295138
2295157	2295159	2295368		2295364	2295142	2295100	2295139
2295243	2295315	2295369		2295365	2295144	2295102	2295303
2295308	2295317	2295370		2322904	2295145	2295147	2295304
2295310	2295319	2295371		2322924	2295232	2295148	2295305
2295312	2295197			2323110	2295268	2295230	2295306
2295314	2295198			2323136	2295269	2295282	2295307
2322874	2295199			2323154	2295270	2295283	2295187
2323053	2295200				2295271	2295284	2295188
2295193					2295272	2295285	2295189
2295194					2295273	2295286	2295190
2295195					2295274	2295287	2295191
2295196					2295281	2295288	2295192
					2322869	2295289	2295254
					2295160	2322867	2295255
					2295161	2295169	
					2295162	2295170	
					2295163	2295171	
					2295164	2295172	
					2295165	2295173	
					2295166	2295174	
					2295167	2295176	
					2295168	2295177	
					2295244	2295248	
					2295245	2295249	
					2295246		
					2295247		

图 9　房间内设备信息(注:第一行为房间 ID)

图 10　房间设备信息 BIM 展示

另外,SQWRL 可以实现对 OWL 本体的查询,找出运维管理所需的信息。示例检索和检索结果如表 4 所示。可以看出,采用 SQWRL 尤其是 built－in 机制的引入可以简便地检索 OWL 本体信息。

表 4　检索示例

检索语句	描述	结果
Q2:tbox:sca(?c,#Room)→sqwrl:select(?c)	所有房间类的个体	图中的房间名
Q3:abox:dpaa(?r,hasRoomID,?i)^hasRoomArea(?r,?a)^hasRoomOccupancy(?r,?o)^swrlm:eval(?l,"a*(100.0-o)/100",?a,?o)→sqwrl:select(?r,?l)	房间可使用面积	所有房间和其可使用面积
Q4:hasRoomName(F1N004,?n)^hasTenancyEndTime(F1N004,?et)^hasRoomArea(F1N004,?a)→sqwrl:select(?n,?a,?et)	某一房间的名称、面积和到期日期	F1N004;77.02;2021－10－01
Q5:Office(?r)^hasRoomArea(?r,?a)^hasRoomOccupancy(?r,?o)^swrlm:eval(?l,"a*(100.0-o)/100",?a,?o)^swrlb:greaterThan(?l, 60.0)→sqwrl:select(?r)	可用面积大于 60m^2 的办公室	F1N004



6　结语

本文建立了基于 BIM 和本体技术的建筑运维管理框架,采用 RevitAPI、OWLAPI 和 SWRLAPI 技术在该框架下建立了一个建筑运维管理系统,采用一个实际案例验证了本文提出方法的有效性。本文提出的系统框架有效集成了 BIM 信息,在客户端上还可实现基于 BIM 的运维管理。个体数据的提取和创建算法可将 TBox 作为 ABox 的信息结构,实现信息在 OWL 知识库的集成。同时在该框架下规则引擎也可方便地被使用。

本文方法为建筑运维管理的语义网实现提供重要基础,对推动 BIM、本体与语义网技术的应用及对建筑运维管理的网络化、智能化应用有重要意义。

参考文献

[1] O'Connor A C, Dettbarn J L, Gilday L T. Cost analysis of inadequate interoperability in the U. S. capital facilities industry[R]. Gaithersburg: NIST, 2004.

[2] Eastman C, Teicholz P, Sacks R, et al. BIM handbook: A guide to building information modeling for owners, managers, designers, engineers, and contractors, 2nd Ed. [M]. Hoboken, NJ: Wiley, 2008.

[3] Buildingsmart. IFC Overview Summary, 2010. http://build – ingsmarttech. org/specifications/ifc-overview.

[4] Venugopal M, Eastman C M, Sacks R, et al. Semantics of model views for information exchanges using the industry foundation class schema[J]. Advanced Engineering Informatics. 2012, 26(2): 411-428.

[5] Gruber T R. Toward principles for the design of ontologies used for knowledge sharing? [J]. International Journal of Human-Computer Studies. 1995, 43(5 – 6): 907-928.

[6] Kontopoulos E, Martinopoulos G, Lazarou D, et al. An ontology-based decision support tool for optimizing domestic solar hot water system selection[J]. Journal of Cleaner Production. 2016, 112, Part 5: 4636-4646.

[7] 陈小平. 语义网基础教程[M]. 北京:机型工业出版社, 2008.

[8] 曾新红, 蔡庆河. OWL2 Web 本体语言文档概述, 2012. http://nkos. lib. szu. edu. cn/OWL2/OWL2OverviewSimplifiedChinese. htm.

[9] O'Connor M, Nyulas C, Shankar R, et al. The SWRLAPI: A Development Environment for Working with SWRL Rules[Z]. Karlsruhe, Germany: 2008.

[10] Horridge M, Bechhofer S. The OWL API: A Java API for OWL Ontologies[J]. Semantic Web journal. 2011, 1(1): 1-11.

[11] O'Conner M, Das A. SQWRL: a Query Language for OWL [Z]. Chantilly, VA: 2009.

[12] Nepal M, Staub-French S, Pottinger R, et al. Ontology-Based Feature Modeling for Construction Information Extraction from a Building Information Model[J]. Journal of Computing in Civil Engineering. 2012, 27(5): 555-569.

[13] B B, F J, N L. Application areas and data requirements for BIM-enabled facilities management[J]. Journal of construction engineering and management. 2011 (3): 431-442.

[14] Bonino D, Castellina E, Corno F. The DOG gateway: enabling ontology-based intelligent domotic environments[J]. IEEE Transactions on Consumer Electronics. 2008: 1656-1664.

[15] Zhang J, El – Gohary N. Automated Extraction of Information from Building Information Models into a Semantic Logic-Based Representation[J]. computing in civil engineering 2015. 2015: 173-180.

A BIM and Ontology-based Approach for the Building Operation and Maintenance Management

Chen Guitao[1,2]

(1. *Shenzhen Institute of Building Research Co., Ltd., Shenzhen* 518049, *China*;
2. *Harbin Institute of Technology Shenzhen Graduate School*, *Shenzhen* 518055, *China*)

Abstract: For the implementation of the building operation and maintenance management on semantic web, a framework for BIM and ontology-based intelligent application is built. Combining some updates with the original DogOnt ontology, a ontology for building operation and maintenance management is created. The methods are proposed to extract information from BIM models for creating individuals and SWRL rules. Proposed framework for BIM and ontology-based intelligent application is implemented by applying the technologies of Revit API, OWL API and SWRL API as Revit built-in programs. The effectiveness of the framework method and programme proposed in this paper are verified by a practical project, and the proposed approach would advance the development of BIM, ontology and semantic web technologies and would be beneficial for networked and intelligent building management of building operation and maintenance.

Key Words: Building Operation and Maintenance Management; Ontology; BIM; OWL; SWRL